

1
2 What is claimed is:

3
4 1. A computer program product including program code, when exe-
5 cuted on a computer system, for providing an interface between a calling ap-
6 plication and at least one callable application, said program code representing
7 a computer program which implements at least two controllers which cooper-
8 ate with each other and are at different hierarchical levels, wherein said con-
9 trollers are instances of a generic controller.

10
11 2. The computer program product of claim 1, wherein the computer
12 program is written in an object-oriented programming language, and the ge-
13 neric controller is a class, and the at least two controllers are subclasses inher-
14 ited from the generic controller class.

15
16 3. The computer program product of claim 1, wherein the calling appli-
17 cation is a computerized business application or an online request handling ap-
18 plication.

19
20 4. The computer program product of claim 1, wherein the at least one
21 callable application is a transaction-tax service application.

22
23 5. The computer program product of claim 4, wherein the at least one
24 transaction-tax service application is a transaction-tax calculation application
25 or a transaction-tax logging application.

26
27 6. The computer program product of claim 4, providing an interface to
28 at least two transaction-tax service applications, said transaction-tax service
29 applications comprising at least two different transaction-tax calculation appli-
30 cations.

31
32 7. The computer program product of claim 1, wherein the controller at

1 the higher hierarchical level is arranged for controlling an overall logic process-
2 ing of the interface, and the controller or controllers at the lower hierarchical
3 level is or are arranged for controlling a processing of the interface specific to
4 the callable application or applications with which the respective controller is
5 associated.

6
7 8. The computer program product of claim 7, wherein
8 the controller at the higher hierarchical level is arranged for receiving an
9 input request from the calling application and sending an output request to the
10 controller at the lower hierarchical level, and receiving an output response
11 from the controller at the lower hierarchical level as an input response,
12 the controller at the lower hierarchical level is arranged for receiving the
13 output request of the controller at the higher hierarchical level as an input re-
14 quest, sending an output request to the callable application or one or more of
15 the callable applications to which it is associated, receiving an input response
16 from the callable application or applications, and sending an output response
17 to the controller at the higher hierarchical level.

18
19 9. The computer program product of claim 8, providing an interface to
20 at least two callable applications of a same type, wherein the specific process-
21 ing for which the controller at the lower hierarchical level and assigned to the
22 at least two callable applications is arranged comprises deciding to which one
23 of the at least two callable applications the output request is sent.

24
25 10. The computer program product of claim 9, wherein the at least two
26 callable applications of the same type are transaction-tax calculation applica-
27 tions.

28
29 11. The computer program product of claim 1, wherein the controllers
30 comprise at least one of the following components:

31 an input/output module;

32 an input parser;

1 a validation engine;
2 a universal state machine;
3 a knowledge base module;
4 a process slip module;
5 a process carrier.

6
7 12. A computer program product including program code, when exe-
8 cuted on a computer system, for providing an interface between a calling ap-
9 plication and at least one callable application, said program code including an
10 interface architecture component, comprising:

11 an input/output module;
12 an input parser;
13 a universal state machine;
14 a knowledge base module;
15 a process slip module;
16 a process carrier.

17
18 13. The computer program product of claim 12, wherein said interface
19 architecture component further comprises a validation engine.

20
21 14. The computer program product of claim 12, wherein said interface
22 comprises at least two cascaded interface architecture components.

23
24 15. A software-implemented method of interfacing a calling application
25 and at least one callable application, comprising:

26 using at least two software-implemented controllers at different hierar-
27 chical levels;

28 performing, both with the controllers at the higher and lower hierarchical
29 levels, a sequence of steps comprising:

30 upon receipt of an input request from an higher hierarchical level element,
31 which is the calling application or a controller at a higher hierarchical level,
32 performing input request handling,

1 sending at least one output request to at least one lower hierarchical level
2 element, which is a controller at a lower hierarchical level or the at least one
3 callable application,

4 receiving an input response to the at least one output request from the
5 lower hierarchical level element,

6 sending an output response to the higher hierarchical level element,

7 wherein the output request of the controller at the higher hierarchical
8 level is the input request to the controller at the lower hierarchical level, and
9 the output response of the controller at the lower hierarchical level is the input
10 response to the controller at the higher hierarchical level.

11
12 16. The method of claim 15, wherein the at least one callable applica-
13 tion is a transaction-tax service application.

14
15 17. The method of claim 15, wherein the at least one transaction-tax
16 service application is a transaction-tax calculation application or a transaction-
17 tax logging application.

18
19 18. A method of implementing a programmed interface between a call-
20 ing application and at least one callable application, comprising:
21 coding at least two controllers at different hierarchical levels, wherein
22 said controllers are instances of a generic controller.

23
24 19. The method of claim 18, wherein the controllers are coded in an ob-
25 ject-oriented programming language, and the generic controller is a class, and
26 the at least two controllers are subclasses inherited from the generic controller
27 class.

28
29 20. The method of claim 18, wherein the at least one callable applica-
30 tion is a transaction-tax service application.

31
32 21. The method of claim 20, wherein the at least one transaction-tax

1 service application is a transaction-tax calculation application or a transaction-
2 tax logging application.

3
4 22. The method of claim 18, wherein the controller at the higher hierar-
5 chical level is arranged for controlling an overall logic processing of the inter-
6 face, and the controller or controllers at the lower hierarchical level is or are
7 arranged for controlling a processing of the interface specific to the callable
8 application or applications with which the respective controller is associated.

9
10 23. The method of claim 22, wherein
11 the controller at the higher hierarchical level is arranged for receiving an
12 input request from the calling application and sending an output request to the
13 controller at the lower hierarchical level, and receiving an output response
14 from the controller at the lower hierarchical level as an input response,

15 the controller at the lower hierarchical level is arranged for receiving the
16 output request of the controller at the higher hierarchical level as an input re-
17 quest, sending an output request to the callable application or one or more of
18 the callable applications to which it is associated, receiving an input response
19 from the callable application or applications, and sending an output response
20 to the controller at the higher hierarchical level.

21
22 24. The method of claim 23, wherein the interface is arranged to pro-
23 vide an interface to at least two callable applications of a same type, wherein
24 the specific processing for which the controller at the lower hierarchical level
25 and assigned to the at least two callable applications is arranged comprises
26 deciding to which one of the at least two callable applications the output re-
27 quest is sent.

28
29 25. The method of claim 24, wherein the at least two callable applica-
30 tions of the same type are transaction-tax calculation applications.

31
32 26. The method of claim 18, wherein the controllers are implemented so

1 as to comprise at least one of the following components:

- 2 an input/output module;
- 3 an input parser;
- 4 a validation engine;
- 5 a universal state machine;
- 6 a knowledge base module;
- 7 a process slip module;
- 8 a process carrier.

9
10 27. A computer program product including program code, when exe-
11 cuted on a computer system, for providing an interface between a calling ap-
12 plication and at least two transaction-tax calculation applications,

13 said interface is arranged to carry out, when called by the calling applica-
14 tion, at least one of:

- 15 - selecting one of the transaction-tax calculation applications depend-
16 ing on a transaction attribute, calling the selected transaction-tax calculation
17 application and receiving a response from the called transaction-tax calculation
18 application; and
- 19 - calling at least two of the transaction-tax calculation applications,
20 comparing the responses returned by them.

21
22 28. The computer program product of claim 27, wherein the interface is
23 further arranged to return a response to the calling application based on the
24 response from the called transaction-tax calculation application or, if at least
25 two transaction-tax calculation applications have been called, based on the
26 comparison.

27
28 29. The computer program product of claim 27, wherein the interface is
29 further arranged to direct a response for logging purposes to a logging control-
30 ler based on the response from the called transaction-tax calculation applica-
31 tion or, if at least two transaction-tax calculation applications have been
32 called, based on the comparison.

1

2 30. A software-implemented method of interfacing a calling application
3 and at least two transaction-tax calculation applications,

4 comprising, when a call is received from the calling application, at least
5 one of:

6 - selecting one of the transaction-tax calculation applications depend-
7 ing on a transaction attribute, calling the selected transaction-tax calculation
8 application and receiving a response from the called transaction-tax calculation
9 application; and

10 - calling at least two of the transaction-tax calculation applications
11 and comparing the responses returned by them.